****

**CERTIFICATE**

**This is to certify that Mr. student of M.Sc(C.S.) Semester III at Suryadatta College of Management Information Research & Technology (SCMIRT), Pune, has successfully completed the assigned practical journal in Advanced Operating System prescribed by the Savitribai Phule Pune University during the academic year 2022-2023.**

**Internal Examiner External Examiner**

**HOD Principal**

**Place: Pune**

**Date: / /**

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.No.** | **Name** | **Page No.** | **Remark** | **Sign** |
| **1)** | **Create an HTML form that contain the Student Registration details and write a JavaScript to validate Student first and last name as it should not contain other than alphabets and age should be between 18 to 50** | **3 - 4** |  |  |
| **2)** | **Create an HTML form that contain the Employee Registration details and write a JavaScript to validate DOB, Joining Date, and Salary** | **5 - 6** |  |  |
| **3)** | **Create an HTML form for Login and write a JavaScript to validate email ID using Regular Expression** | **7 - 8** |  |  |
| **4)** | **Create a Node.js file that will convert the output "Hello World!" into upper-case letters:** | **9** |  |  |
| **5)** | **Using nodejs create a web page to read two file names from user and append contents of first file into second file** | **10 - 11** |  |  |
| **6)** | **Create a Node.js file that opens the requested file and returns the content to the client. If anything goes wrong, throw a 404 error** | **12 - 13** |  |  |
| **7)** | **Create a Node.js file that writes an HTML form, with an upload field** | **14** |  |  |
| **8)** | **Create a Simple Web Server using node js** | **15** |  |  |
| **9)** | **Using node js create a User Login System** | **16 - 18** |  |  |
| **10)** | **write node js script to interact with the filesystem, and serve a web page from a file** | **19** |  |  |

1. **Create an HTML form that contain the Student Registration details and write a JavaScript to validate Student first and last name as it should not contain other than alphabets and age should be between 18 to 50.**

**Ans:**

**Prac1.html:**

<!DOCTYPE html>

<html>

<head>

<title>Slip 1 Name and Age Validation</title>

<script src="Slip1.js"></script>

</head>

<body>

<form> First Name:

<input type="text" id="fname"></input></br></br> Last Name:

<input type="text" id="lname"></input></br></br> Age:

<input type="text" id="age"></input></br></br>

<button type="submit" onclick="validateForm()">Submit</button>

<input type="text" id="output1"></input>

</form>

</body>

</html>

**Slip1.js:**

function validateForm() {

var fname = document.getElementById("fname").value;

var lname = document.getElementById("lname").value;

var age = document.getElementById("age").value;

var reg = /^[a-zA-Z]+$/;

if (fname.length == 0 || lname.length == 0 || age.length == 0) {

alert("All Fields are Mandatory");

return false;

}

else if (!reg.test(fname) || !reg.test(lname)) {

alert("Only Alphabets Allowed");

return false;

}

else if (age < 18 || age > 50) {

alert("Enter Age between 18 & 50 Only")

return false;

}

else {

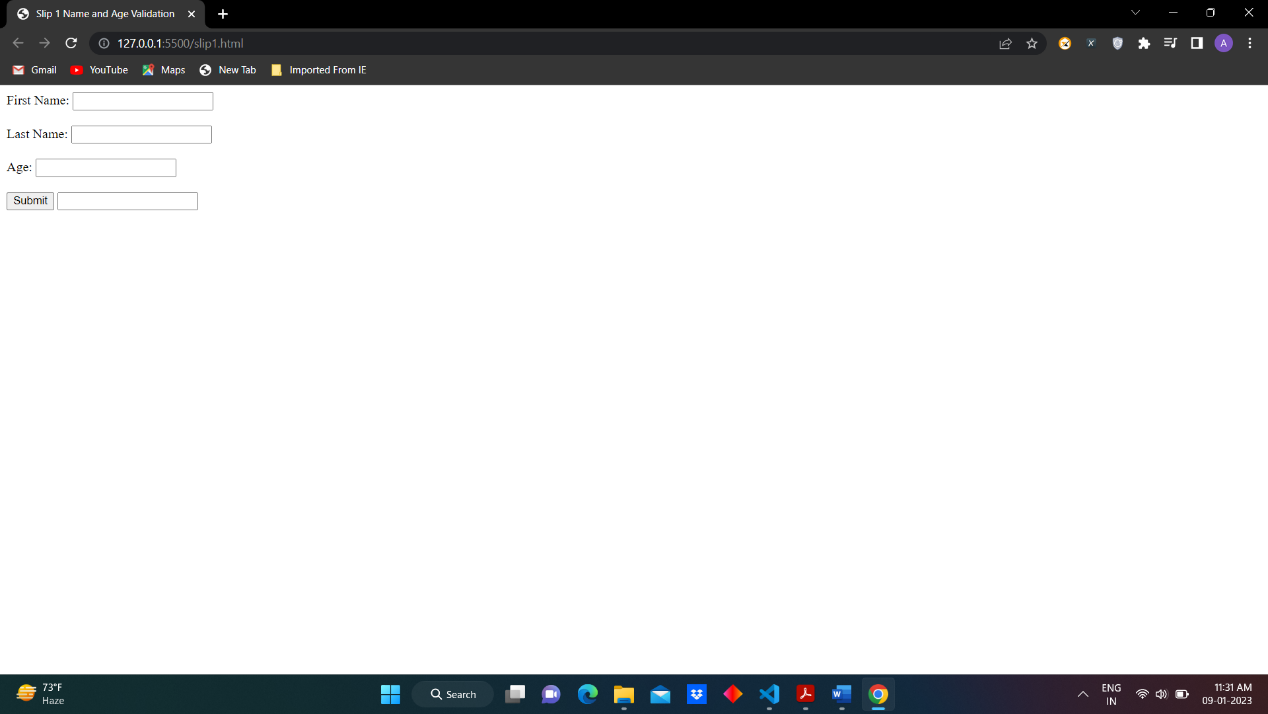
document.getElementById('output1').value = age + " " + fname + " " + lname;

return true;

}

}

**Output:**



1. **Create an HTML form that contain the Employee Registration details and write a JavaScript to validate DOB, Joining Date, and Salary.**

**Ans:**

**prac2.html:**

<!DOCTYPE html>

<html>

<head>

<title>Slip 2 Employee Registration Details</title>

<script src="Slip2.js"></script>

</head>

<body>

<form> Full Name: <input type="text" id="name"></input></br></br> DOB <input type="text" id="dob"></input></br></br>

Joining Date <input type="text" id="jdate"></input></br></br>

Salary <input type="text" id="sal"></input></br></br> <button type="submit"

onclick="validateform()">Submit</button> </form>

</body>

</html>

**Slip.js:**

function validateform() {

var name = document.getElementById("name").value;

var dob = document.getElementById("dob").value;

var jdate = document.getElementById("jdate").value;

var sal = document.getElementById("sal").value;

var regName = /^[a-zA-Z0-9]+\s[a-zA-Z0-9]+$/;

var regdate = /^[0-9]{1,2}\/[0-9]{1,2}\/[0-9]{4}$/;

var regsal = /^[0-9]$/;

if (name.length == 0 || dob.length == 0 || jdate.length == "" || sal.length == 0) {

alert("All Fields are Mandatory");

return false;

}

else if (!regName.test(name)) {

alert("Enter Name Format Correctly First Name Last Name");

return false;

}

else if (!regdate.test(dob)) {

alert("Registration Date Format DD/MM/YYYY");

return false;

}

else if (!regdate.test(jdate)) {

alert("Date of Joining Format DD/MM/YYYY");

return false;

}

else if (!regsal.test(sal)) {

alert("Enter Numerical Values only in Salary");

return false;

} else {

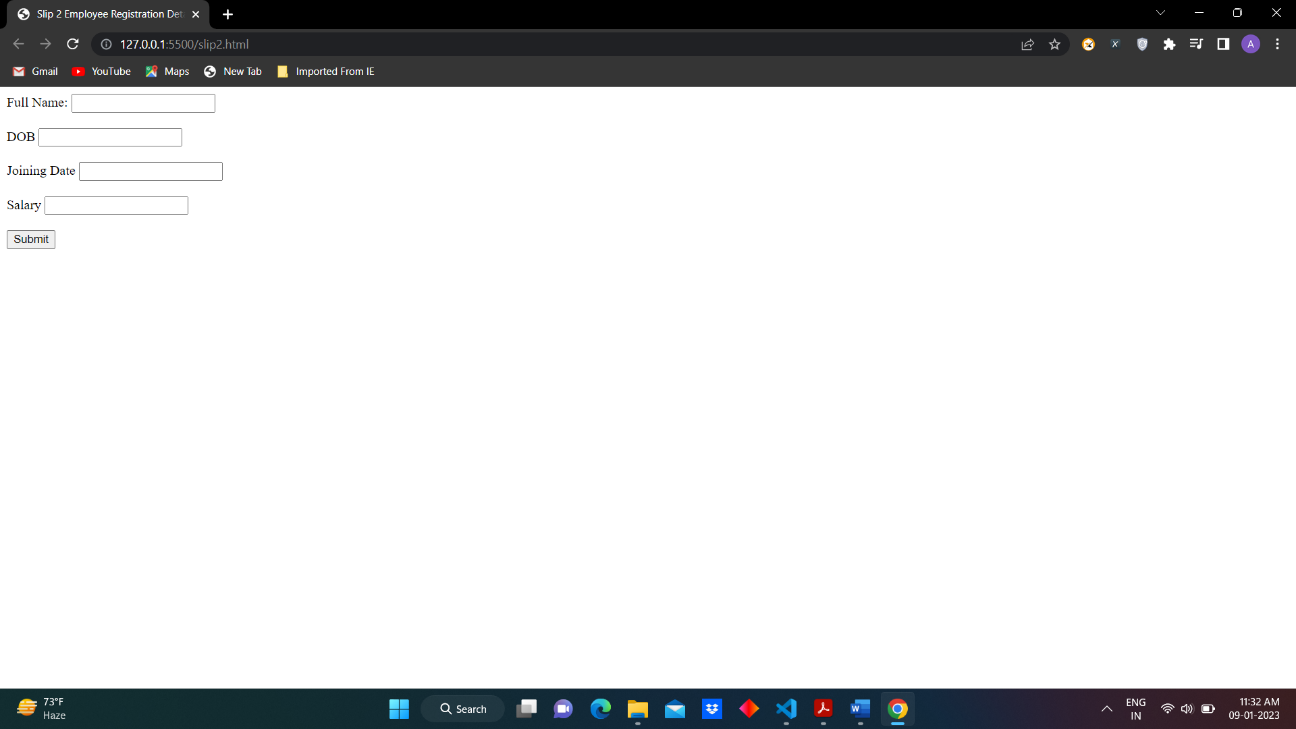
alert("Validation Successfull")

return true;

}

}

**Output:**



1. **Create an HTML form for Login and write a JavaScript to validate email ID using Regular Expression.**

**Ans:**

**prac3.html:**

<!DOCTYPE html>

<html>

<head>

<title>Slip 3 Email Validation</title>

<script src="Slip3.js"></script>

</head>

<body>

<form> Email: <input type="text" id="email"></input></br></br> Password: <input type="password"

id="pass"></input></br></br> <button type="submit" onclick="validateform()">Submit</button></form>

</body>

</html>

**Slip3.js:**

function validateform() {

var email = document.getElementById("email").value;

var pass = document.getElementById("pass").value;

var regEmail = /^([a-zA-Z0-9\.\_]+)@([a-z]+)(.[a-z]+)?$/;

if (email.length == 0) {

alert("All Fields are Mandatory");

return false;

}

else if (pass.length == 0) {

alert("All Fields are Mandatory");

return false;

}

else if (!regEmail.test(email)) {

alert("Enter Name Format Correctly First Name Last Name");

return false;

}

else {

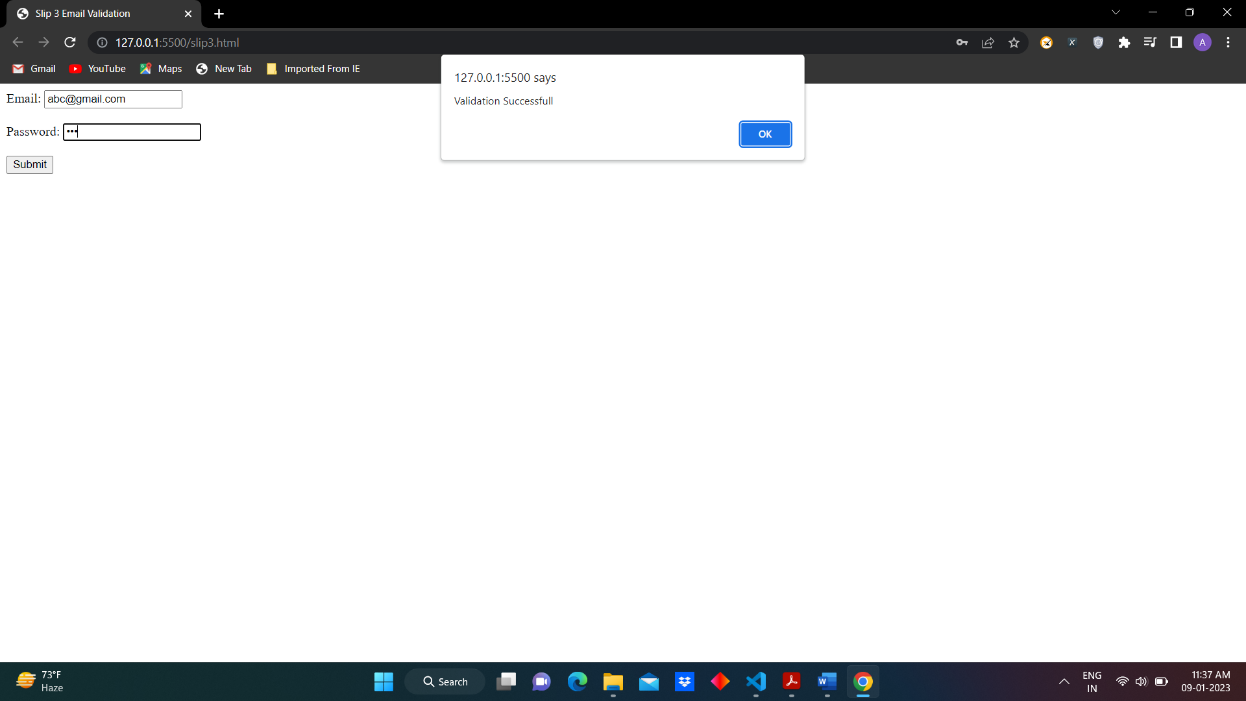
alert("Validation Successfull");

return true;

}

}

**Output:**



1. **Create a Node.js file that will convert the output "Hello World!" into upper-case letters:**

**Ans:**

var http = require('http');

var uc = require('upper-case');

http.createServer(function (req, res) {

res.writeHead(200, {'Content-Type': 'text/html'});

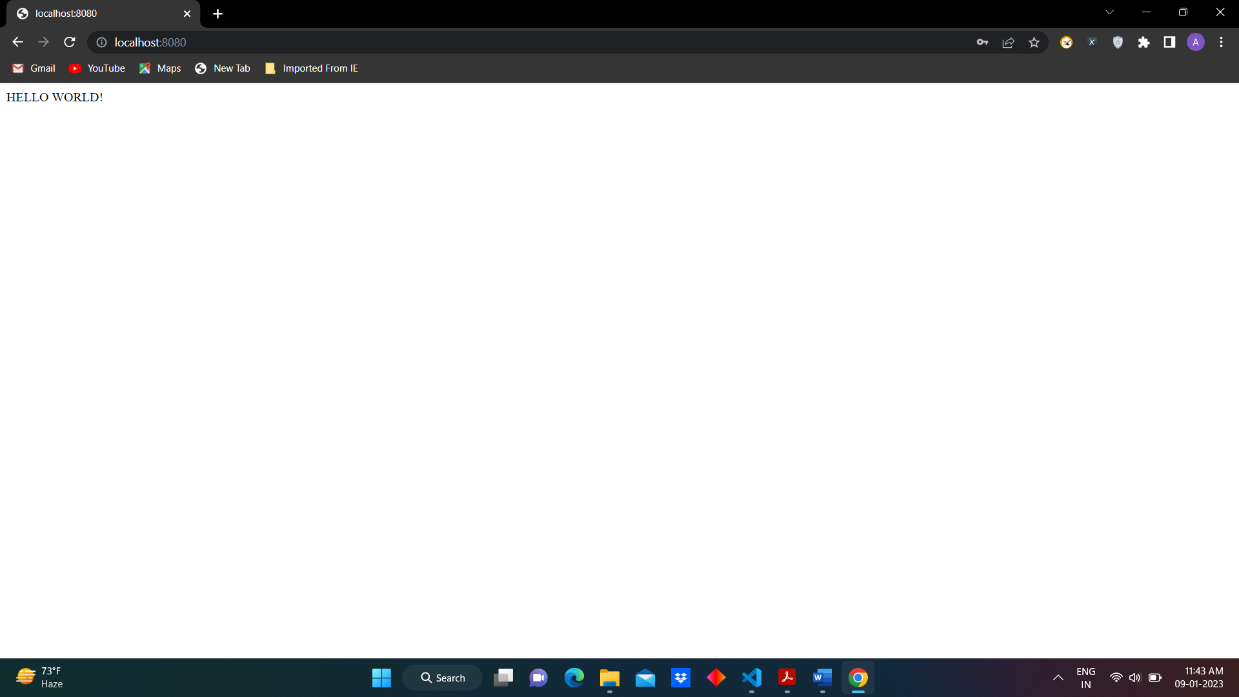
/\*Use our upper-case module to upper case a string:\*/

res.write(uc.upperCase("Hello World!"));

res.end();

}).listen(8080);

**Output:**



1. **Using nodejs create a web page to read two file names from user and append contents of first file into second file**

**Ans:**

var fs = require('fs');

var path = require('path')

var data = fs.readFileSync("Sample1.txt", "utf8");

fs.appendFile("Sample2.txt", data, (err) => {

if (err) {

console.log(err);

}

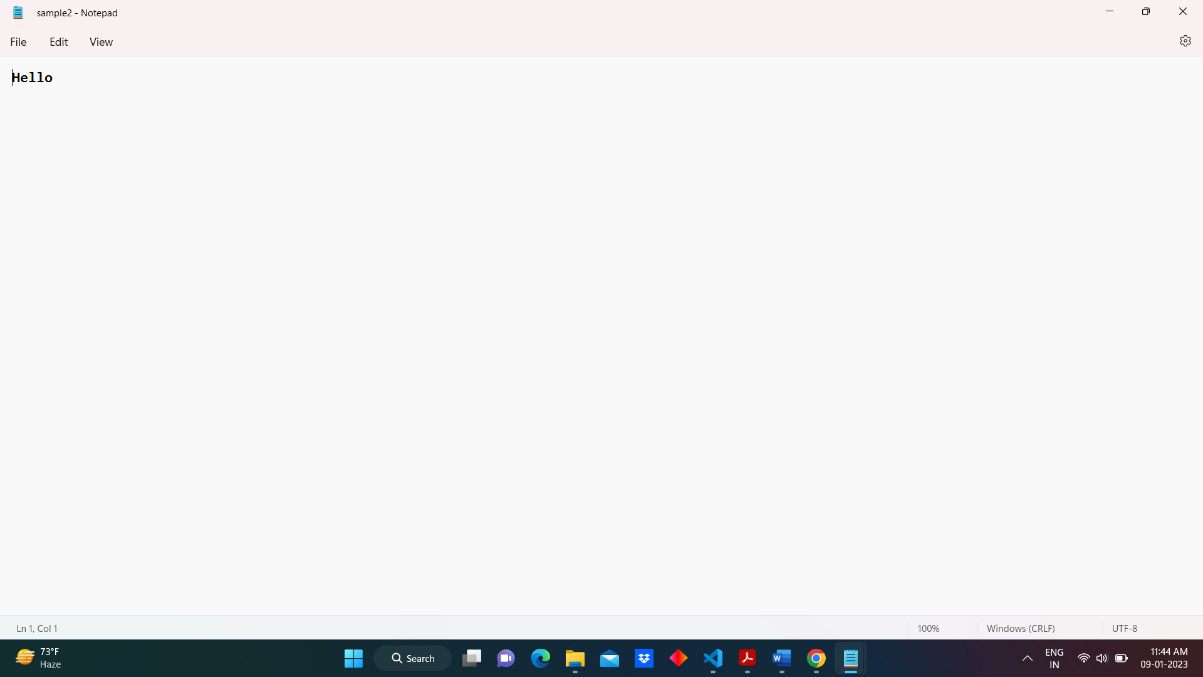
else {

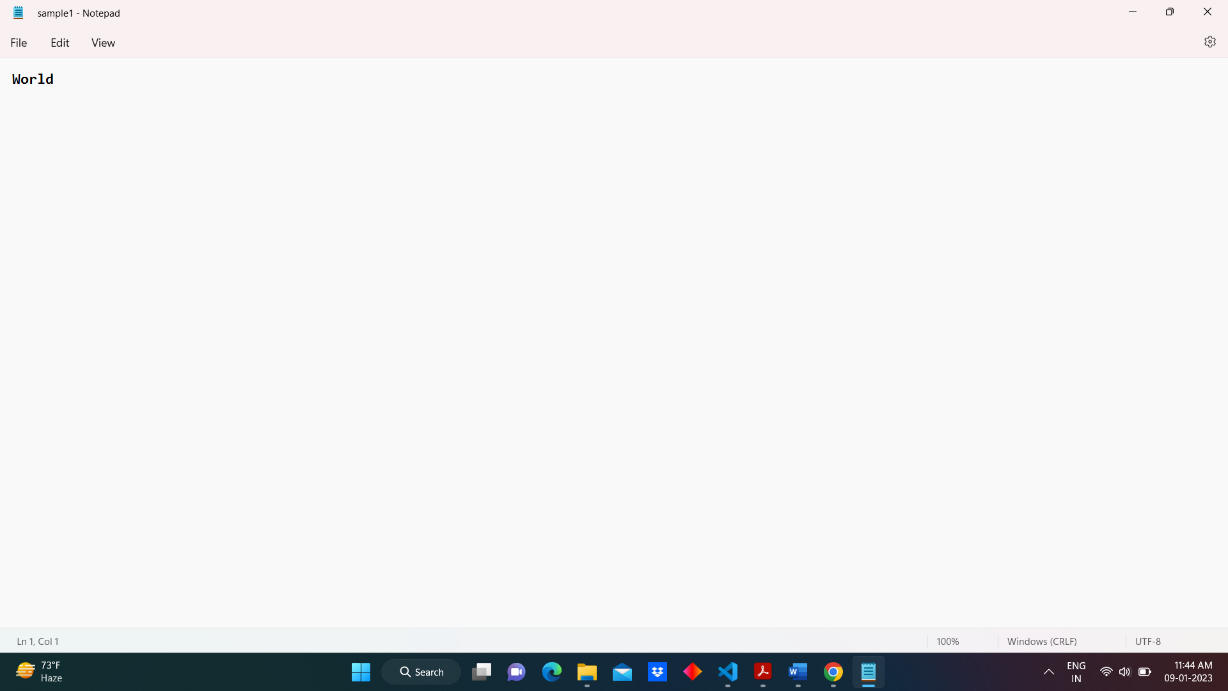
console.log("File Content after appending: ", fs.readFileSync("Sample2.txt", "utf8"));

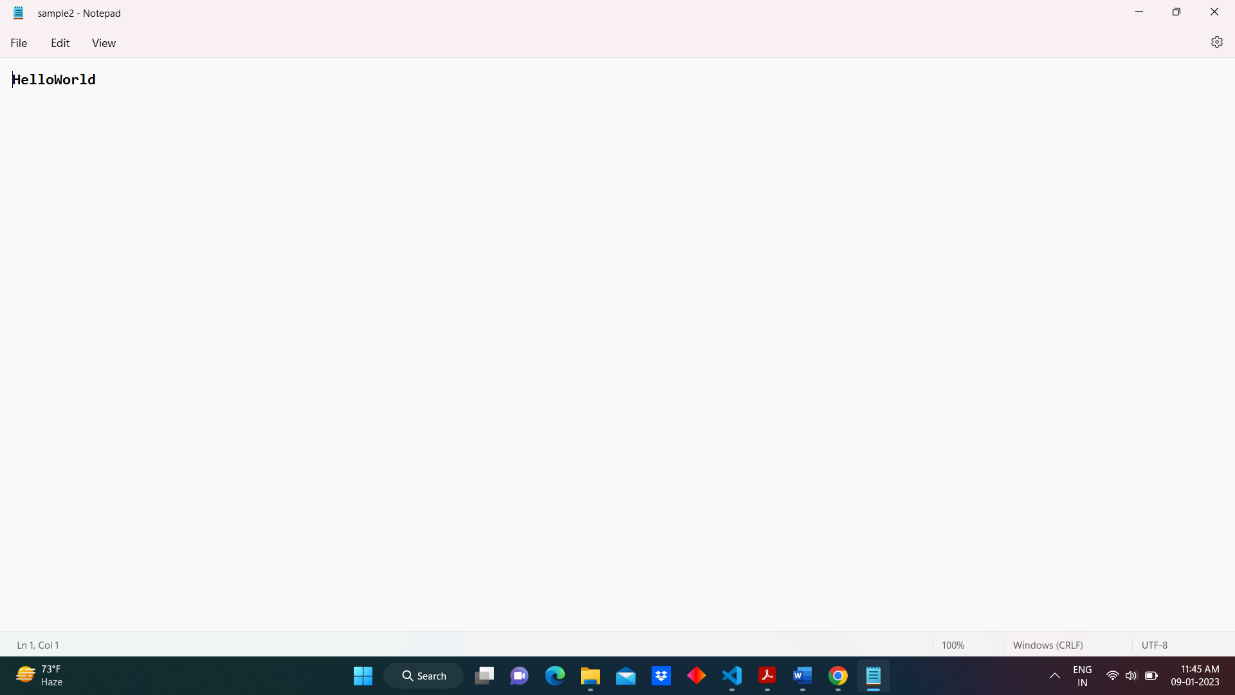
}

});

**Output:**







1. **Create a Node.js file that opens the requested file and returns the content to the client. If anything goes wrong, throw a 404 error**

**Ans:**

var http = require('http'); var fs = require('fs');

var server = http.createServer(function (req, res) {

fs.open('Sample.txt', 'r', function (err, fd) {

if (err) {

res.writeHead(404, { 'Content-Type': 'text/html' });

return res.end("404 File Not Found");

}

else {

console.log("File Open Successfully");

fs.readFile('Sample.txt', function (err, data) {

if (!err) {

console.log('File Read Successfully'); res.end(data); fs.close(fd);

}

else {

console.log('Read File is not possible');

}

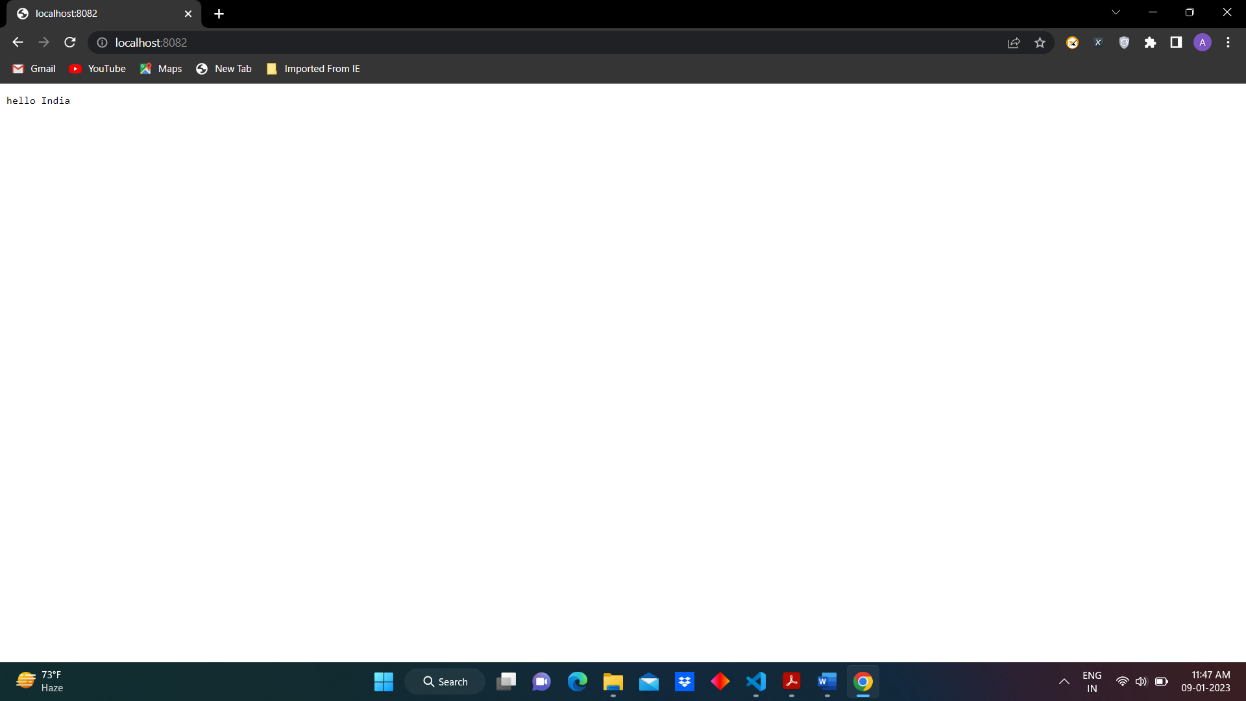
});

}

});

}).listen(8081);

**Output:**



1. **Create a Node.js file that writes an HTML form, with an upload field**

**Ans:**

var http = require('http');

var formidable = require('formidable');

http.createServer(function (req, res) {

var form = new formidable.IncomingForm();

form.parse(req, function (err, fields, files) {

if (req.url == '/fileupload') {

console.log(files);

res.write('File Uploaded');

res.end();

}

else {

res.writeHead(200, { 'Content-Type': 'text/html' });

res.write('<form action = "fileupload" method = "get" enctype = "multipart/form\_data">');

res.write('<input type = "file" name="fileuploaded"><br><br>');

res.write('<input type = "submit">');

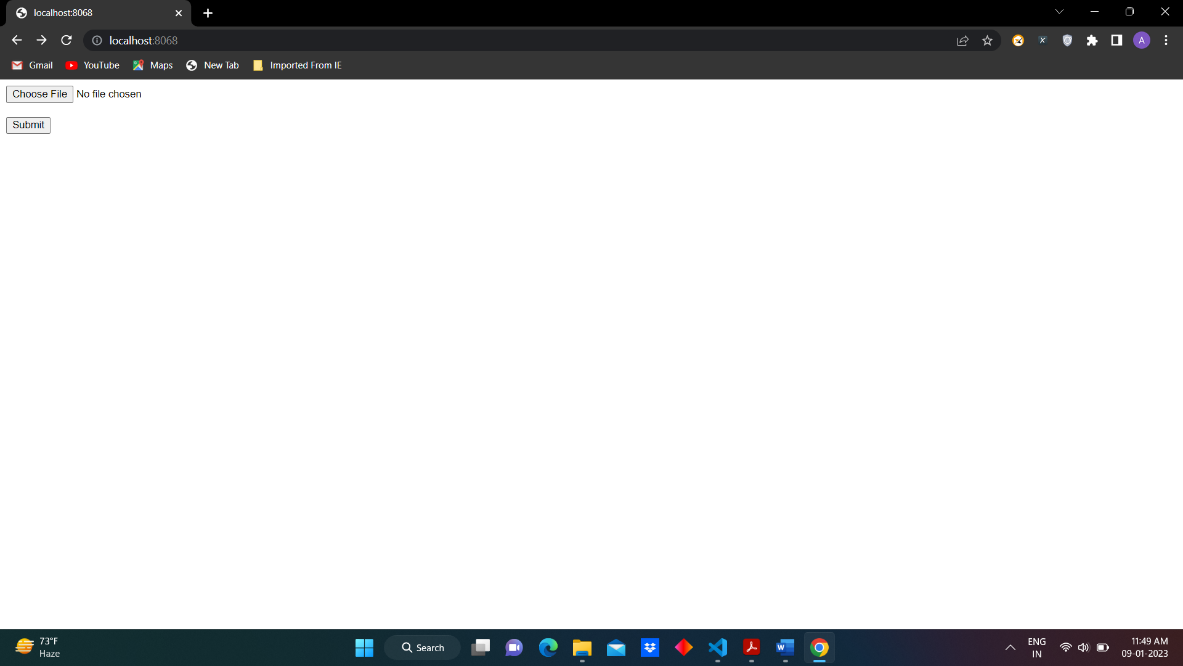
res.write('</form>'); return res.end();

}

});

}).listen(8068);

**Output:**



1. **Create a Simple Web Server using node js**

**Ans:**

var http = require('http');

http.createServer(function(req,res){

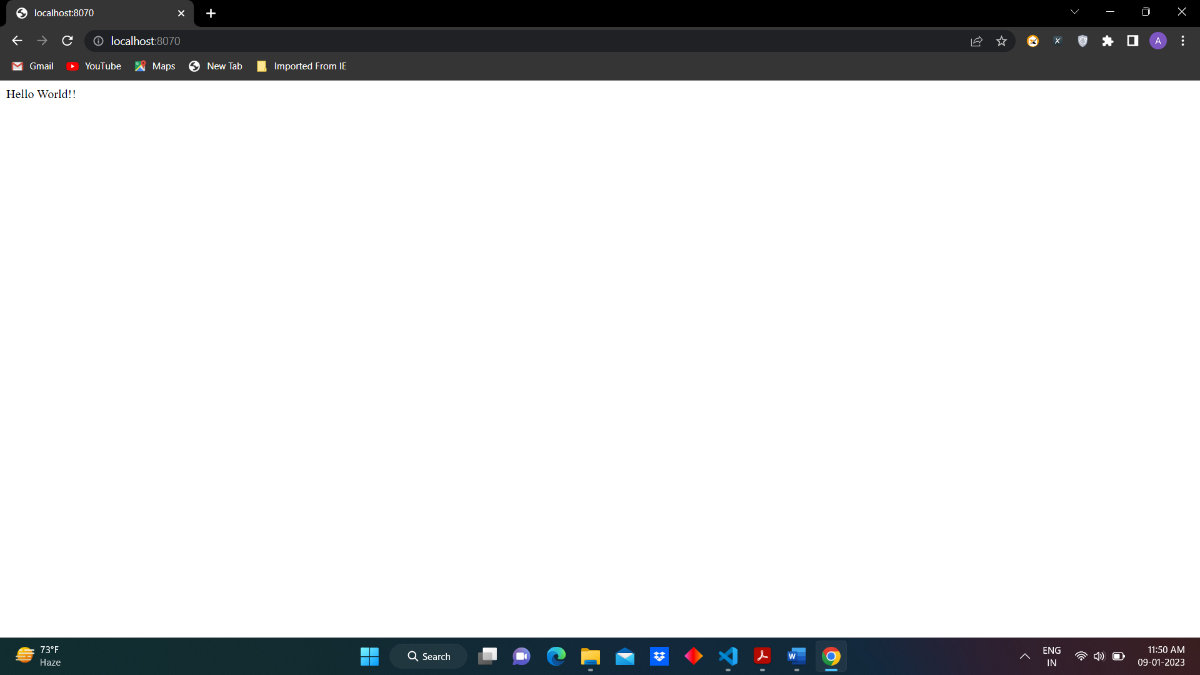
res.writeHead(200, {'Content-Type':'text/html'});

res.write("Hello World!!");

res.end();

}).listen(8070);

**Output:**



1. **Using node js create a User Login System**

**Ans:**

**Index.html:**

<html>

<head>

<title>Welcome Home</title>

</head>

<body>

<a href="http://localhost:3000/login">Please Login Here!</a>

</body>

</html>

**Login.html:**

<html>

<head>

<title>LOGIN FORM</title>

</head>

<body>

<form action="http://localhost:3000/login" method="post"> Username: <br> <input type="text" name="usrname" required>

<br><br> Password: <br> <input type="password" name="pass" required> <br><br>

<input type="submit" value="LOGIN">

</form>

</body>

</html>

**Login.js:**

var express = require('express');

var bodyParser = require('body-parser');

var app = express(); app.use(bodyParser.urlencoded({ extended: false }));

app.get('/', function (req, res) { res.sendFile(\_\_dirname + '/index.html'); });

app.get('/login', function (req, res) { res.sendFile(\_\_dirname + '/login.html'); });

app.post('/login', function (req, res) {

person = { username: req.body.usrname, password: req.body.pass };

res.end("Logged in Successfully " + JSON.stringify(person));

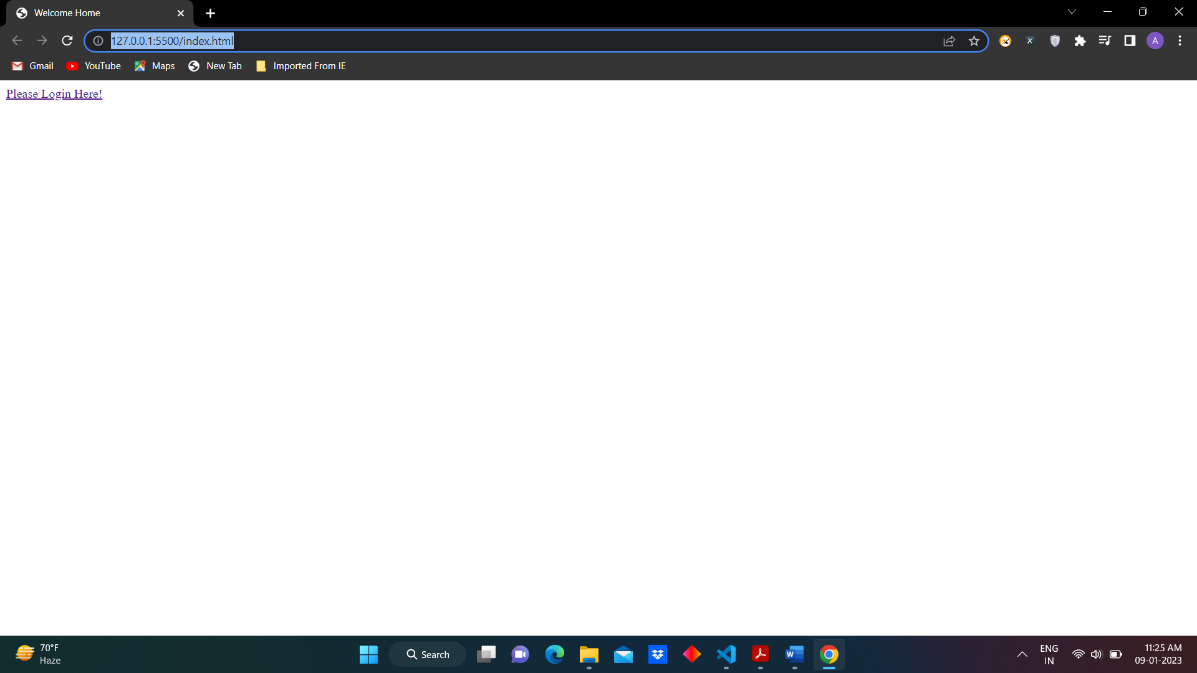
});

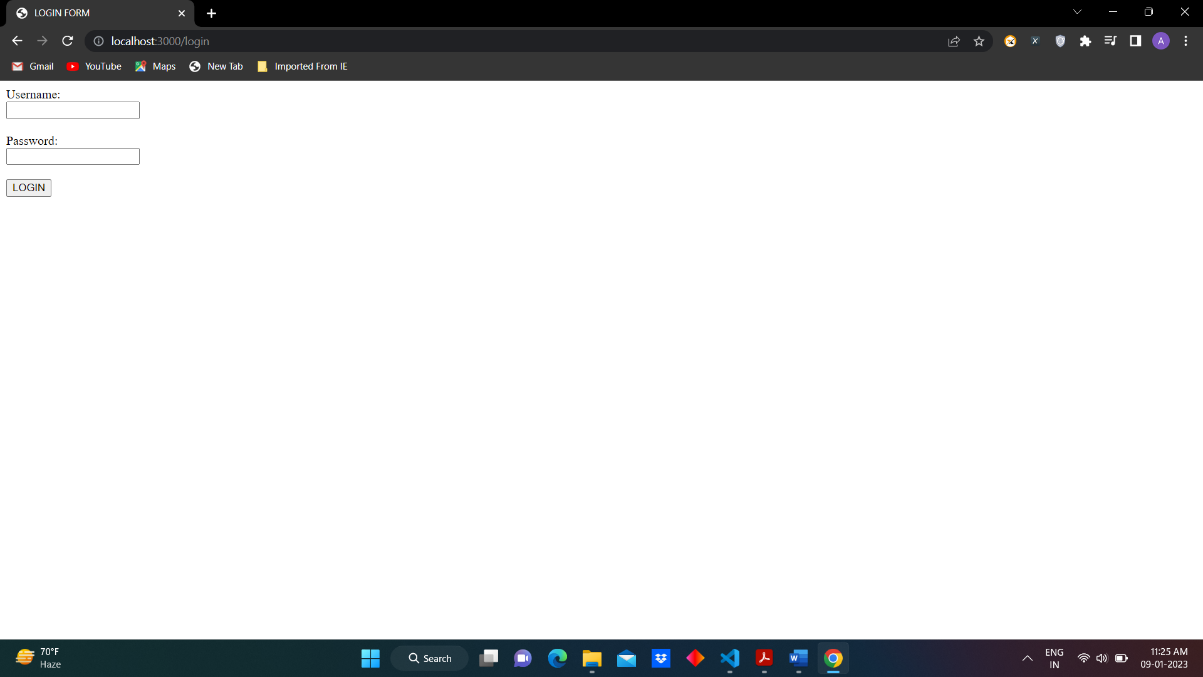
var server = app.listen(3000, function () {

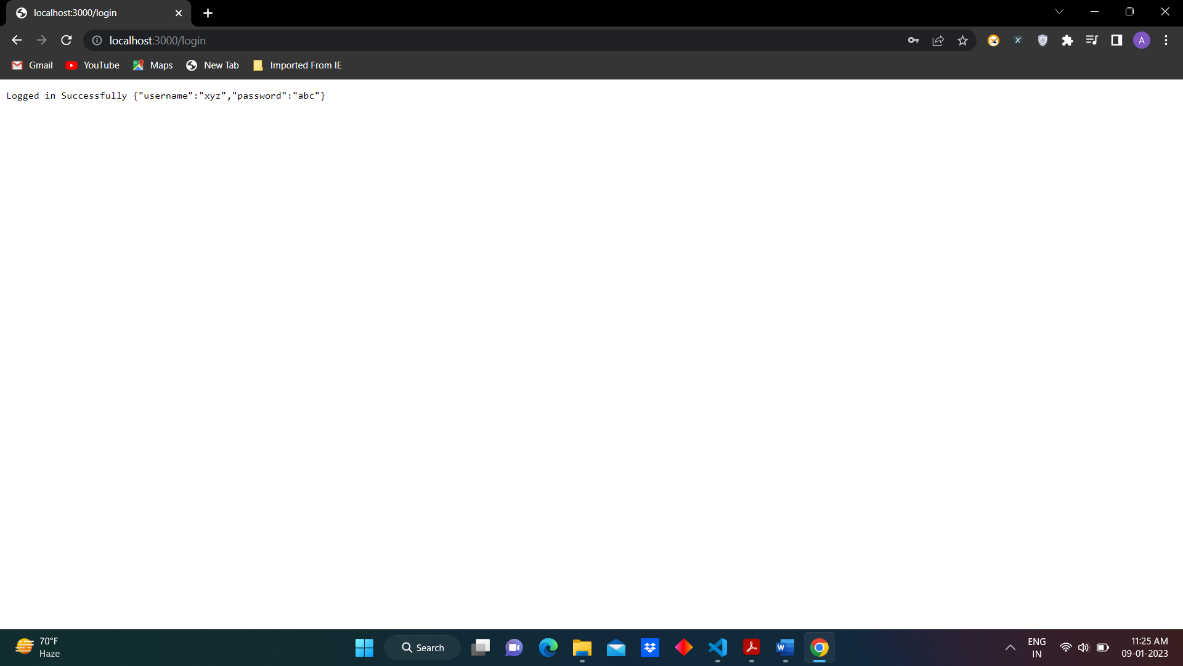
console.log("Server is Running");

});

**Output:**







**10) write node js script to interact with the filesystem, and serve a web page from a file**

**Ans:**

var http = require('http');

var fs = require('fs');

var server = http.createServer(function (req, res) {

//fs.open( filename, , mode[r = read & r+ = readwrite], callback )

fs.open('Sample.txt', 'r', function (err, fd) {

fs.readFile('Sample.txt', function (err, data) {

if (!err) {

console.log('File Read Successfully');

res.end(data);

fs.close(fd);

}

else {

console.log('Read File is not possible');

}

});

});

}).listen(8060);

**Output:**

